

# Framework para o Desenvolvimento de Organizações de Sistemas Multi-agentes com Distribuição Autônoma de Tarefas

Leandro Ramos da Silva<sup>1</sup>, Márcia Ito<sup>2</sup>

<sup>1</sup>Laboratório de Pesquisa em Ciência de Serviços/Centro Paula Souza– São Paulo – Brasil

<sup>1</sup>[lramos@gmail.com](mailto:lramos@gmail.com), <sup>2</sup>[marcia.ito@centropaulasouza.sp.gov.br](mailto:marcia.ito@centropaulasouza.sp.gov.br)

**Resumo** - Sistemas Multi-gentes é uma área de estudo da Inteligência Artificial Distribuída cuja implementação demanda técnicas e ferramentas que facilitem lidar com a complexidade inerente de sistemas com alto processamento paralelo. O presente trabalho apresenta o desenvolvimento de organizações de agentes gerenciadas por um *broker* e distribuição autônoma das tarefas que visa reduzir a complexidade para adoção de sistemas multi-agentes. Uma simulação de monitoramento de pacientes crônicos e os resultados são apresentados neste artigo, demonstrando a eficácia da distribuição independente de tarefas para determinar um número de agentes necessários em uma sociedade.

Palavras-chave: Sistemas Multiagentes, SMA, Organização.

**Abstract** - Multi-agent systems it's a Distributed Artificial Intelligence study area whose implementation requires techniques and tools that facilitate dealing with the inherent complexity of such systems with high-parallel processing. This paper presents the agent organizations' development managed by *broker* and autonomous tasks distribution to reduce the complexity for the adoption of multi-agent systems. A simulation of chronic patients monitoring and the results are presented in this paper, demonstrating the effectiveness of independent tasks distribution to determine a number of agents needed in a society.

Keywords: Multi-agent systems, MAS, Organization.

## 1. Introdução

Nota-se pela diversidade de técnicas (BALDONI; BOELLA; GENOVESE; GRENNAN; TORRE, 2008), métodos (HANNOUN; BOISSIER; SICHMAN, 2000; SCHWAMBACH, 2004) e ferramentas (BELLIFEMINE, 2003) para aplicação em Sistemas Multi-agentes que não há uma teoria única comumente aceita

acerca de agentes, e por esta razão ao se deparar com uma implementação prática nesta categoria de sistemas torna-se necessário adaptar algumas técnicas.

Motivando-se por essas questões, foi desenvolvida esta solução que agrupa algumas técnicas, métodos e ferramentas para criação de um *framework de racionalização*, e de um *Broker* (para agenciar agentes). A função dos agentes na sociedade pode mudar com o tempo, dependendo do conjunto de tarefas que ele assume autonomamente. Nesta solução para os agentes se organizarem, devem negociar entre si e ter um agente gerente denominado *broker*.

Neste *framework* instituem-se agentes distribuídos em organizações gerenciadas, nas quais é requerido o aspecto conversacional dos agentes e sua capacidade de simular raciocínio sobre *objetivos* através da seqüência linear ou não de *ações* mantendo sempre o estado do ambiente, sem a necessidade da utilização de uma linguagem de programação, e sim por meio de regras pré-definidas pelos administradores.

## **2. Framework para Organizações de SMAs**

O *framework* proposto suporta a simulação de capacidade de raciocínio que seriam alimentadas através de regras pré-definidas, e principalmente implementa a distribuição dinâmica dos agentes incluindo a criação de novos agentes para execução de novas atividades, pois considera-se tendo em vista o panorama de Sistemas Multi-agentes que a organização entre os agentes é um ponto crucial para o sucesso do modelo, tendo em vista que o conhecimento coletivo dos Agentes o qual proporciona a correta interação com os usuários, e não os mesmos individualmente.

A categorização de SMAs é difícil sem o auxílio dos projetistas e desenvolvedores do programa, já que isso depende da aplicação e recursos necessários ao software de agentes sendo desenvolvido. (WOOLDRIDGE, 2000).

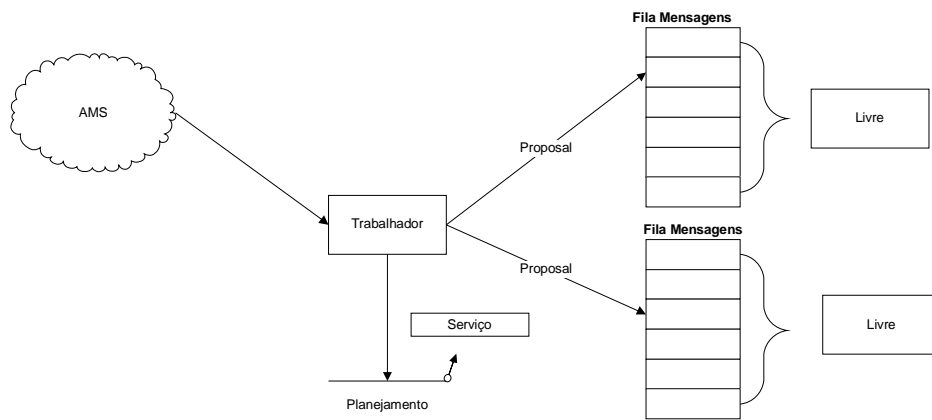
A partir então da classificação de sistemas de Durfee (1994) onde de acordo com as características dos agentes (se são fixas ou variáveis) e do ambiente (se são fixas ou variáveis), propõe-se uma categoria com propriedades de agentes *variáveis* e propriedades de ambiente *variáveis*, retomando ainda Rosenschein e Genesereth (1985) que definem essa categoria como RDP que são uma sub-categoria de SMA onde os agentes são benevolentes, ou seja, “*deseja* ajudar-se mutuamente sempre que possível” (DURFEE, 1994, p. 4), um dos aspectos principais da negociação de tarefas proposta. Então na solução proposta os agentes ficam em uma posição de atingir a objetivos específicos sem saber o quanto determinado objetivo vai ou não agradá-lo.

## 2.2. Distribuição de tarefas na sociedade

O mecanismo central de organização das sociedades de agentes instituídas nesta arquitetura é a possibilidade dos agentes assumirem novas tarefas, através de um diálogo baseado em CNET (**Contract-net**).

Esta proposta não possui representação formal da sociedade deixando com que os agentes se agrupem em um SMA de acordo com as necessidades, ao estilo das organizações implicitamente definidas do tipo AC definidas por Jaime em (HUBNER; SICHMAN, 2003): os agentes se organizam de acordo com suas próprias motivações e cada agente pode ter uma representação da sociedade diferente das demais.

Os agentes se organizam através da emissão de uma proposta para algum agente livre atender a determinado serviço, comportamento que é configurável e cadastrável representado na figura 1.



Elf t q 0 , @ dmsd Aqj dqd O' o, lr 'RHKU@+1//7(

## 3. Resultados

Algumas simulações foram realizadas no domínio do acompanhamento de pacientes crônicos, o modelo GRPC (ITO, 2006), no entanto, como o objetivo não é comprovar o modelo GRPC nem a funcionalidade “Monitorar níveis glicêmicos”, foram implementadas somente duas cenas “Informar resultado glicemia” e “analisar resultado” que por si só envolvem três agentes em negociação. Portanto, os papéis dos agentes são aqueles descritos em (ITO, 2006):

- Atendente: encaminhar para “Coletar Glicemia” e mostrar o menu principal;

- Enfermeiro: “Coletar glicemia”: Abrir ocorrência, perguntar data da medição, perguntar hora da medição, perguntar valor da medição e encaminhar para “Analisar o resultado”;
- Diabetologista: “Analisar resultado”: Analisar glicemia e emitir laudo.

A finalidade desta simulação é averiguar a eficácia da distribuição automática de papéis. Isto é possível ao comparar o framework com um modelo estabelecido, no caso, um modelo na qual não há gerencia na criação dinâmica de novos agentes.

Desta forma, para a aplicação dos testes algumas adaptações foram necessárias para tornar uma comparação da proposta viável. Além disso a constante evolução de hardware e criação de novos dispositivos, as configurações exatas das máquinas tornam-se irrelevantes e não são consideradas para as simulações deste trabalho.

Primeiramente foi realizada uma simulação para compreender os efeitos de muitos usuários simultâneos e o impacto na organização do tempo que um usuário leva entre ler uma pergunta, digitar e enviar a resposta.

Número de atendimentos	Número de agentes	Resposta usuário	Tempo total de atendimento
15	3	1ms	2s
<b>15</b>	3	<b>1000ms</b>	<b>34s</b>
200	3	1ms	12s
<b>200</b>	3	<b>1000ms</b>	<b>5m42s</b>

S` adk 0 | Qdrt kš cnr nasčnr m rH t k žŸn bnl n eq / dv nq hm šun d sq,r  
` f dnšdr 'RHKU@+1/17(

Pela tabela nota-se que o fato de que quanto maior o número de atendimentos (15 e 200) ou o tempo de resposta do usuário, maior o tempo total de atendimento por parte de todos os agentes. Constata-se com isso a suspeita de que o funcionamento da organização está diretamente relacionado ao número de usuários e de sua interação individual com os agentes.

Na tabela 2 o mesmo problema foi tratado, desta vez com o *framework ativo* na qual os agentes se organizam em 55 agentes realizando os mesmos 200 atendimentos (com 1000ms de tempo de resposta do usuário) em 27s.

Número de atendimentos	Número de agentes	Resposta usuário	Tempo total de atendimento
<b>15</b>	11	<b>1000ms</b>	<b>10s</b>
200	13	1ms	11s
<b>200</b>	55	<b>1000ms</b>	<b>27s</b>

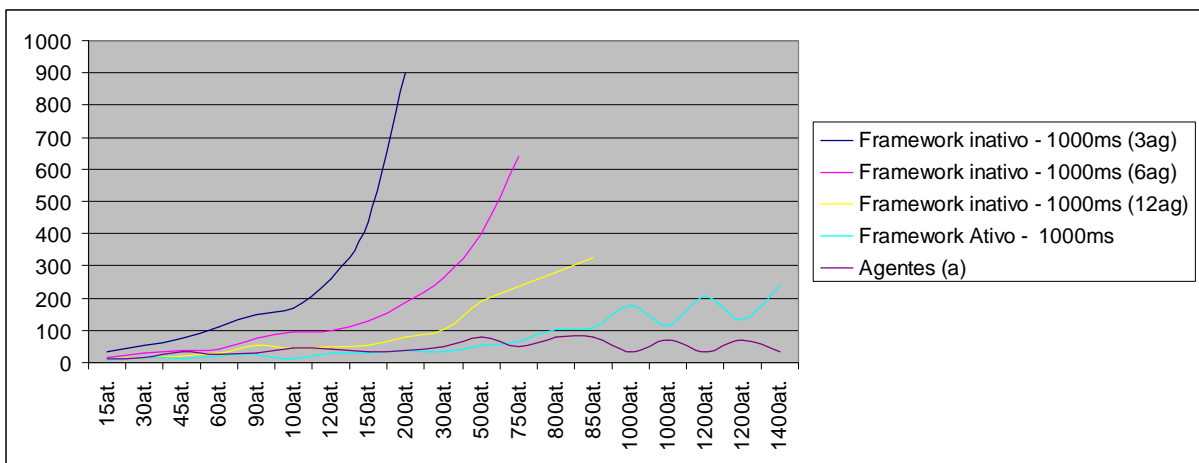
S` adk 1 | Qdrt kè cnr nasçnr m` rH t k ÷ÿn bnl n eçl dv nç ` sun 'RHKU@-1//7(

A partir da comparação dos testes discriminados na tabela 1 (*framework inativo*) com os testes discriminados na tabela 2 (*framework ativo*), um resultado chamou muito a atenção: o teste realizado para 200 atendimentos com 1000ms de tempo de resposta por cada pergunta emitida ao usuário. O *framework ativo* (tabela 2: 55 agentes em 27s) realizou os mesmos 200 atendimentos com quase 2000% a mais do número de agentes do *framework inativo* (tabela 1: 3 agentes em 5m42s), porém em 8% do tempo deste.

Porém, o teste seguinte ainda não era conclusivo se não apresentasse um estudo mais aprofundado a cerca do número de agentes criado pelo *framewor ativo*. Uma simulação então considerando 1000ms (1 segundo) para o usuário ler uma pergunta e emitir uma resposta foi apresentado ao *framework ativo*, e em seguida o *framework inativo* com 3 agentes (um diabetologista, um enfermeiro e um atendente). Posteriormente, mais duas simulações foram realizadas: com o *framework inativo* utilizando 6 e 12 (tabela 3) agentes respectivamente distribuídos igualmente entre si, gerando o gráfico 5-2:

Número de atendimentos	Número de agentes	Tempo total de atendimento	Framework ativo	Tempo resposta
200	12	41s	Sim	1ms
200	12	1min 24s	Não	1ms

S` adk 2 | Bnl o` ç sun dmsçl n eçl dv nç ` sun d n eçl dv nç lm sun 'l ' pt lm cd l , çm cdr dl odmgn('RHKU@-1//7(

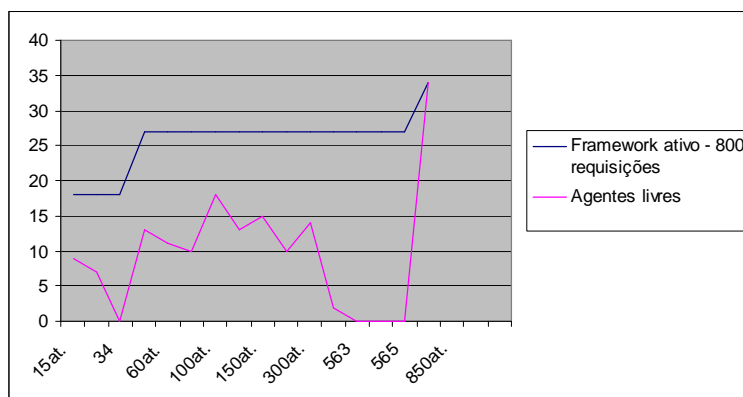


F ç dbn 0 , Dunk ÷ÿn cd sdl on cd ` sçmçH dmsn bnl o` ç su` dmsçl` rH t k ÷ÿn m eçl dv nç ` sun d m eçl dv nç lm sun 'çl onr s` r cn t rt ` çn dl t l r df t mçn('RHKU@-1//7(

Ainda assim, independentemente da redução de 50% do tempo de execução total da sociedade quando os agentes assumem tarefas dinamicamente comparadas a uma distribuição manual das tarefas, deve-se ressaltar que os testes foram feitos no *framework inativo* sabendo-se o número de agentes de antemão obtidos de testes com o *framework* ativo e fazendo a distribuição manual das tarefas. Pode-se com isso demonstrar que a distribuição de tarefas automática é útil para se identificar a quantidade ótima de agentes na sociedade e para que eles organizem-se de forma autônoma, sem intervenção humana.

#### 4. Conclusões e Trabalhos Futuros

Conforme indicado no Gráfico 2 o número de agentes total é incrementado de acordo com a necessidade de agentes livres para o trabalho, não considerando nenhum estudo sobre gerenciamento do hardware.



F q d b n 1 , F d q l r b h l d m n c n m l d q n c d ` f d m s d r k u q p r m e q l d v n q ` s h u n  
' b n l 7 / / q p t l r h t O d r m d k ( ' R H K U @ + 1 / / 7 (

Além disso, o Gráfico 2 apresenta uma queda inicial de agentes livres que é imediatamente suprida já que o número de agentes total é incrementado. Como o gráfico apresenta um vale extenso quando há um número maior de agentes trabalhando na sociedade isso nos leva a crer que a maior demanda por recursos de máquina impulsionada por um maior número de agentes trabalhando faz com que o agente *broker* tenha dificuldades na criação de novos agentes para ajudar a sociedade.

Uma possibilidade é onde o Broker possa antes de criar um novo agente solicitá-lo a algum gerenciador de GRID, que é uma aplicação que permite a um grande range de aplicações paralelas executarem em ambientes distribuídos se utilizando da capacidade de hardware disponível. Isto é possibilitado integrando máquinas clientes de usuários e recursos em laboratórios compartilhados em uma intranet ou ambiente Internet baseada em Computação em Grid (FOSTER; KESSELMAN, 1999).

Um exemplo de gerenciador de grades é o Integrate (GOLDCHLEGER; KON; GOLDMAN; FINGER; BEZERRA, 2004), um middleware que permite a

criação de grades computacionais oportunistas. Ele é organizado como “uma federação de aglomerados, onde cada aglomerado contém máquinas que disponibilizam seus recursos ociosos para utilização por aplicações da grade.” (GOLDCHLEGER; KON; GOLDMAN; FINGER; BEZERRA, 2004, p. 2).

Um requisito importante do InteGrade é que os usuários decidem compartilhar suas máquinas sem o Grid perceber ou impactar a qualidade do serviço provido por suas aplicações. A arquitetura do InteGrade foi cuidadosamente projetada com este requisito em mente. Assim, máquinas clientes utilizam uma implementação leve em CORBA e o acesso aos recursos de hardware são cuidadosamente controlados por um agendador em nível de usuário, permitindo ainda que aplicações paralelas que tipicamente executem sob a forma dedicada compartilhem recursos (GOLDCHLEGER; KON; GOLDMAN; FINGER; BEZERRA, 2004). Além disso, sua arquitetura possui um componente que monitora o padrão de uso dos componentes do Grid, o que minimiza a imprevisibilidade de um ambiente tão dinâmico.

A aplicação dessa tecnologia no *framework* proposto é a integração do mecanismo *broker* ao conceito de *Global Resource Manager* (GRM) de forma a orientar as estações participantes sobre o local mais adequado (fisicamente) onde os agentes devem ser introduzidos na sociedade.

## Referências

- HAIT, F. B. **Agent oriented programming with Ada'95: application to financial markets**. Volume XX , Issue 1 (March 2000) table of contents. Pages: 67 – 80. Year of Publication: 2000. ISSN:1094-3641. Université Paris XII., 61, avenue du GAL de Gaulle. 94010 Creteil Cedex Publisher ACM Press New York, NY, USA
- BASS, L.; CLEMENTS, P.; KAZMAN, R. **Software architecture in practice**. SEI – Series in Software Engineering. 2. ed. Addison Wesley, 2003. 528 p.
- BALDONI, M.; BOELLA, G.; Genovese, V.; GRENN, R.; van der TORRE L.; How to Program Organizations and Roles in the JADE Framework. In: GERMAN CONFERENCE ON MULTI-AGENT SYSTEM TECHNOLOGIES, MATES 2008, September 2008, Germany. R. Bergmann, G. Lindemann, S. Kirn, and M. Pechoucek, eds, **Proceedings of the Multiagent System Technologies**, volume 5244, Germany, 2008. p. 25-36
- BELLIFEMINE F.; CAIRE, G.; TRUCCO, T.; RIMASSA, G. **JADE Programmer's Guide, 2003**. Disponível em <http://sharon.cselt.it/projects/jade/doc/programmersguide.pdf>
- BOELLA, G.; TORRE, L. van der. Attributing mental attitudes to roles: the agent metaphor applied to e-trade organizations. In: 6TH INTERNATIONAL CONFERENCE ON ELECTRONIC COMMERCE. Session: Multi-agent systems and social behavior table of contents. Delft, The Netherlands. **ACM**

**International Conference Proceeding Series.** Vol. 60. ISBN:1-58113-930-6. Università di Torino, Italy. p. 130 - 137. 2004.

BOELLA, G.; TORRE, L. van der. Regulative and constitutive norms in normative multiagent systems. In: INTERNATIONAL CONFERENCE ON PRINCIPLES OF KNOWLEDGE REPRESENTATION AND REASONING (KR'04), 9, 2004, Whistler, Canada. **Proceedings of KR'04.** AAAI Press, 2004.

p. 255—265.

BOELLA, G.; TORRE, L. van der. Organizations as Socially Constructed Agents in the Agent Oriented Paradigm. [Engineering Societies in the Agents World V](#) 1-13. Book Series [Lecture Notes in Computer Science](#) Publisher Springer Berlin / Heidelberg ISSN0302-9743 (Print) 1611-3349 (Online) Volume 3451/2005 Book Copyright 2005

DURFEE, E. H.; ROSENSCHEIN, J. S. Distributed Problem Solving and Multi-Agent Systems: Comparisons and Examples. In: 13TH INTERNATIONAL WORKSHOP ON DISTRIBUTED ARTIFICIAL INTELLIGENCE, IWDAI-94. Seattle, 1994. **Proceedings.** Seattle, AAI, 1994. p. 46-55.

FOSTER, I.; KESSELMAN, C. **The Grid: Blueprint for a New Computing Infrastructure.** Morgan Kaufmann Publishers: San Francisco, 1999.

FRANKLIN, S.; GRAESSER, A. Is it an Agent, or just a Program?: A Taxonomy for Autonomous Agents. In: 3RD. INTERNATIONAL WORKSHOP ON AGENT THEORIES, ARCHITECTURES, AND LANGUAGES. **Proceedings of the 3rd. International Workshop on Agent Theories, Architectures, and Languages.** Springer-Verlag, 1996.

GOLDCHLEGER, A.; KON, F.; GOLDMAN, A.; FINGER, M.; BEZERRA, G.C. InteGrade: object-oriented Grid middleware leveraging idle computing power of desktop machines. In **Concurrency and Computation: Practice and Experience.** Middleware for Grid Computing. Volume 16, Pages 449 – 459. 26 Mar 2004

HANNOUN, M.; BOISSIER, O.; SICHMAN, J.S.; SAYETTAT, C. MOISE: An organizational model for multi-agent systems. In: BRAZILIAN SYMPOSIUM ON AI (IBERAMIA-SBIA 2000), International joint conference No15, Atibaia, BRASIL, (19/11/2000). **Proceedings of International joint conference.** ISBN 3-540-41276-X. Atibaia, BRASIL, 2000, vol. 1952, pp. 156-165[Note(s) : XV, 498 p., ] (14 ref.)

HUBNER, J.F.; SICHMAN, J.S. Organização de Sistemas Multiagentes In II Jornada de Atualização em Inteligência Artificial (in Portuguese), JAIA 2003, XXI Congresso da Sociedade Brasileira de Computação, SBC'2003, em CD, 40 pages, Campinas, Brazil, August 2003. Disponível em <http://www.lti.pcs.usp.br/moise/doc/orgSMA-jaia-2003.pdf>

ITO, Márcia. Um modelo de gestão de paciente crônico baseado nos conceitos de relacionamento com o cliente. 2006. p. 19 (Motivações e resultados esperados), p. 30 (Principais sistemas de monitoramento de pacientes



- crônicos). Tese (Doutorado) - Escola Politécnica da Universidade de São Paulo, São Paulo, 2006.
- MEA, V. D. Agents acting and moving in healthcare scenario: a paradigm for telemedical collaboration. IEEE Transactions on Information Technology in Biomedicine, (s. l.), v. 5, no. 1, p. 10-13. March 2001.
- ROSENSCHEIN, J.S.; GENESERETH, M.R. Deals among rational agents. In: 9TH INTERNATIONAL JOINT CONFERENCE ON AI (IJCAI 1985), 9, Los Angeles, California, USA, 1985. **Proceedings of the 9th International Joint Conference on AI**. Los Angeles, CA: Morgan Kaufmann, 1985. p. 91-99.
- URUGUAY, A., HIRATA, C. Using IDEF0 to Enhance Functional Analysis in MOISE+ Organizational Modeling. Brazilian Artificial Intelligence Symposium, 2006
- SCHWAMBACH, M. M.; PEZZIN, J.; FALBO, R. A. OplA: uma metodologia para o desenvolvimento de sistemas baseados em agentes e objetos. In: JORNADA IBEROAMERICANA DE ENGENHARIA DE SOFTWARE E ENGENHARIA DO CONHECIMENTO, 4, Madrid, 2004. **Proceedings**.
- SILVA, L. R. da. **Desenvolvimento de um framework para o gerenciamento da organização de agentes em sistemas multi-agentes**. 96 f. + apêndices. Dissertação (Mestrado) - Centro Estadual de Educação Tecnológica Paula Souza. São Paulo: CEETEPS, 2008.
- VENKATESAN, V.P. Arquitetura para orquestração de serviços com agente BDI. White Paper, Microsoft. Disponível em: [http://www.microsoft.com/brasil/msdn/arquitetura/20080428/Arquitetura\\_agente\\_BDI.mspix](http://www.microsoft.com/brasil/msdn/arquitetura/20080428/Arquitetura_agente_BDI.mspix). Acessado em 15/10/2008

Leandro Ramos da Silva

Centro de Estadual de Educação Tecnológica Paula Souza

Unidade de Pós-Graduação, Extensão e Pesquisa

Laboratório de Pesquisa em Ciência de Serviços (LaPCiS)

R. Bandeirantes, 169 - Bom Retiro - 01124-010 - São Paulo - SP - Brasil  
tel/fax: (11) 3327-3104

e-mail: [lramos@gmail.com](mailto:lramos@gmail.com)

Márcia Ito

Centro de Estadual de Educação Tecnológica Paula Souza

Unidade de Pós-Graduação, Extensão e Pesquisa

Laboratório de Pesquisa em Ciência de Serviços (LaPCiS)

R. Bandeirantes, 169 - Bom Retiro - 01124-010 - São Paulo - SP - Brasil  
tel/fax: (11) 3327-3104

e-mail: [marcia.ito@centropaulasouza.sp.gov.br](mailto:marcia.ito@centropaulasouza.sp.gov.br)